REMARKS

Claims 1-20 were pending in the application. Claim 21 was added. Various claim amendments were made. Therefore, Claims 1-9, 12-21 are presented for examination herein.

Response to Claim Rejections – 35 USC 103

Claims 1-9, 12-20 have been amended and Claim 21 added. Claim 21 is the same as the un-amended Claim 1 from the previous amendment. Support for the new claim aspects can be found, among other places, at: P13: L 13-30, P14: L17-27, P15: L17-27, P17: L11, P18: L24.

Note that Claims 1-9, 12-20 have been amended such that any combination of Liddy (US6,304,864) and Adams (6,334,145) would not result in the applicants claimed invention. This is because Adams does not teach the concept of performing a multilevel linked search over a set of documents where the search is defined in terms of the Domain boundary as opposed to a number of links in depth and/or breadth. Therefore Claims 1-9, 12-20 are in condition for allowance.

Claim 21 is the same as Claim 1 from the previous amendment. However, Claim 21 is in condition for allowance for the reasons outlined below. That is, Liddy teaches away from a combination of Liddy and Adams. Liddy expressly teaches away from Claim 21 and thus all the amended claims would be allowable without amendments. However, the Amendments have been made because it is deemed that the presently claimed embodiment would be the most useful. Applicant therefore reserves the right to pursue material similar to Claim 21 in a continuation.

Liddy (US6,304,864) teaches a system that uses remote agents to perform web searches. Liddy teaches to cast a search net as wide as possible to find results that might not be reachable from a particular search engine. By sending out agents to both perform web crawling and to plug parameters into search engines, a very expansive and broad web search can be performed to find otherwise difficult to find results.

However, Liddy teaches away from the concept of placing restrictions on the number of links to be followed. This is because Liddy is interested in broad Internet searches for any and all documents that meet a certain criterion. This is in contrast to the

currently claimed aspects of the present invention which use exportable software agents to search locally around a target web page. For example, one might be interested in looking for all PDF files that discuss TMS320C50 DSPs, but only those found on the Texas Instruments web site, only those returned by a particular search engine, or only those within a set number of links from a page that is close to having the desired information.

Instead, Liddy teaches in the opposite direction, i.e., to a system where web crawler agents continue crawling until they run out of links to follow. In fact, a small army of web crawlers coordinated by a leader agent are sent out to search for specific types of projects. The leader agent locally manages the other agents who store intermediate address information in a client-side local memory 19. See especially Liddy at Col. 9, line 49 to Col 10, line 65. Note that Col. 9, line 49 to Col 10, line 65 of Liddy teach away from the claimed feature of the invention that allows a user to drill deeper in a controlled manner based on the results from a previous search or to search all linked documents within the boundaries of a specified domain.

Liddy at Col. 10, line 31 states: "**No restrictions need be placed on the number of levels of linked documents from the document at the first Web address.** If any crawler agent locates multiple link addresses in a document, the address is temporarily stored in a queue in memory 19 until the same of another crawler agent is available to retrieve a document from the WWW associated with that address."

Liddy at Col. 10, line states: "The agent leader can reuse existing crawler agents which have **stopped crawling due to all addresses linked to their stating address have been retrieved.**"

Because Liddy teaches to send out an army of specially trained (neural network trained) agents to search far and wide across the internet for specified information, there is no motivation to add to Liddy the limited linked multilevel searching capabilities of Adams (6,334,145) to arrive at the Applicant's invention. This is because such a combination would defeat the purpose of Liddy, which is to perform an expansive search, much wider than would even be cast by a major search engine (see Liddy at "Summary of the Invention").

That is, the Applicant's invention, as currently claimed, teaches a system that allows localized areas of web space to be characterized in fine detail, for example to drill deeper on current search results, to create a site map of a specified domain, or to find all linked documents within certain boundaries based on domain boundaries or number-of-hops types of distance measures. The aspects present invention as claimed herein produce a significantly different result than the result produced by Liddy.

Liddy, on the other hand, teaches in an opposite direction away from the current invention. One of ordinary skill in the art, without the hindsight of the present invention disclosure, would not have been motivated to combine Liddy with Adams. This is because Liddy expressly stated there is no need to do this, and also because doing so would violate the purpose of Liddy which is to search in as many areas as possible to locate all documents, to include hard-to-find documents, and to filter the set of all found documents using a neural network based filtering algorithm.

Summary

Claims 1-9, 12-21 were presented for examination herein.

Applicant notes that any amendments or claim cancellations made herein and not substantively discussed above are made solely for the purposes of more clearly and particularly describing and claiming the invention, and not for purposes of overcoming art. The Examiner should infer no (i) adoption of a position with respect to patentability, (ii) change in the Applicant's position with respect to any claim or subject matter of the invention, or (iii) acquiescence in any way to any position taken by the Examiner, based on such amendments or cancellations not substantively discussed.

Furthermore, any remarks made herein with respect to a given claim or amendment are intended only in the context of that specific claim or amendment, and should not be applied to other claims, amendments, or aspects of Applicant's invention.

Applicant specifically reserves the right to prosecute claims of differing and broader scope than those presented herein in a continuation application.

Lastly, Applicant notes that any amendments made by this paper which are not specifically discussed herein are made solely for the purpose of more clearly and particularly pointing out and claiming Applicant's invention.

If the Examiner has any questions or comments which may be resolved over the telephone, he is requested to call the undersigned at (305) 735-8533, fax a message at 305-437-7670, or send an email to dsperic@aol.com.

Respectfully submitted,

Eric M. Dowling

Dated: December 3, 2004

By: 

Eric M. Dowling
Registration No. 44,094
Interlink 731
PO Box 02-5635
Miami, FL 33102-5635
Tel: (305) 735-8533
Fax: (305) 437-7670

EMD/Mult.001-CIP1
12/3/04

IN THE CLAIMS:

Please **ADD** claim 21. Please AMEND the pending claims as indicated below:

1. (Presently Amended) A ~~smart browser~~ client-side software module comprising:

~~an application layer interface coupled to a protocol stack, said application layer interface operative to receive at least one data packet comprising at least a portion of a target web page;~~

a user interface for interacting with a user and for identifying a target web page;

a multilevel search control interface configured to accept at least two parameters, one that defines a content based search string and another that ~~defines a linked set of documents that each have a hyperlink linkage of a specified range of linkage levels referenced from the target web~~ page; identifies a set of linked set of documents that are linked via one or more levels of hyperlinks to the target web page;

a ~~set of~~ portable executable program code that is responsive to the at least two parameters, and is programmed to perform a multilevel search to search each of the documents specified in the linked set of documents for the content that matches the content based search string, wherein ~~documents~~ each document in the linked set of documents ~~are characterized in that they can be reached from the target through a sequence of N hyperlink activations, where N is a fixed integer;~~ has an address within a specified domain and can be reached from the target document via a sequence of hyperlinks that are each found in documents in the linked set of documents;

a network based program module that causes at least a portion of the portable set of portable executable program code to be transmitted to a server that is adapted to accept executable code and to execute such code on behalf of one or more remote clients;

wherein ~~upon being received at the server,~~ at least some executable code that has been transferred from the browser module is permitted to execute at the server, and upon execution, the portion of the executable code that is executed orchestrates a multilevel search to cause the documents in the linked set of documents to be analyzed to determine whether they match the content based search string.

2. (Presently Amended) The ~~browser~~ client-side software module of Claim 1, wherein said user interface comprises a window display providing an interactive menu to a user.

3. (Presently Amended) The ~~browser~~ client-side software module of Claim 2, wherein said user window is a part of a windows based graphical user interface.

4. (Presently Amended) The ~~browser~~ client-side software module of Claim 1, ~~where said user interface comprises a voice interface.~~ wherein each file found in the search is automatically downloaded by the client-side software module to create a local linked representation of the linked set of documents.

5. (Presently Amended) The ~~browser~~ client-side software module of Claim 1, ~~wherein said multilevel browser operation corresponds to a multilevel "find in page" operation.~~ wherein each document found in the search is downloaded and a local sitemap representative of the linking structure of the linked set of documents is created by the client-side software module.

6. (Presently Amended) ~~A multilevel search browser plug-in module for coupling to a host browser, whereby the host browser comprises a markup language parser, and a user interface for coupling to a user, and an application layer communications interface, said application layer interface operative to receive at least one data packet comprising at least a portion of a target web page the plug-in module comprising:~~

~~a multilevel search control interface a set of portable executable program code that is responsive to the at least two parameters, and is programmed to perform a multilevel search to search each of the documents specified in the linked set of documents for the content that matches the content based search string, wherein documents in the linked set of documents are characterized in that they can be reached from the target through a sequence of N hyperlink activations, where N is a fixed integer;~~

~~a network based program module that causes at least a portion of the portable set of portable executable program code to be transmitted to a server that is adapted to accept executable code and to execute such code on behalf of one or more remote clients;~~

~~wherein upon being received at the server, at least some executable code that has been transferred from the browser module is permitted to execute at the server, and upon~~

~~execution, the portion of the executable code that is executed orchestrates a multilevel search to cause the documents in the linked set of documents to be analyzed to determine whether they match the content based search string.~~

A client-side software module comprising:

a user interface for interacting with a user and for identifying a target web page;

a multilevel search control interface configured to accept at least three parameters to include i) one that defines a content based search string and ii) one that identifies a set of linked set of documents that are linked via one or more levels of hyperlinks to the target web page and iii) one that specifies one or more file types as defined by one or more file name extensions;

a portable executable program code that is responsive to the at least two parameters, and is programmed to perform a multilevel search to search each of the documents specified in the linked set of documents for the content that matches the content based search string, wherein each document in the linked set of documents has an address within a specified domain and can be reached from the target document via a sequence of hyperlinks that are each found in documents in the linked set of documents;

a network based program module that causes at least a portion of the portable set of portable executable program code to be transmitted to a server that is adapted to accept executable code and to execute such code on behalf of one or more remote clients;

wherein at least some executable code that has been transferred from the browser module is permitted to execute at the server, and upon execution, the portion of the executable code that is executed orchestrates a multilevel search to cause the documents in the linked set of documents to be analyzed to determine whether they match the content based search string.

7. (Presently Amended) The plug-in module of Claim 6, wherein said ~~plug-in module~~ client-side software module ~~is embodied as Java™ code.~~ is operative to search for all files with an extension of .ppt within the specified domain.

8. (Presently Amended) The plug-in module of Claim 6, wherein said ~~plug-in module~~ client-side software module ~~is embodied as executable XML code.~~ is operative to download all files found in the search.

9. (Presently Amended) For use in a client ~~browser~~ computerized device, a method comprising the steps of:

obtaining application data from an application layer interface;

passing said information to a user via a user interface;

coupling a multilevel-search interface signal to a user, the multilevel search interface being configured to accept at least two parameters, one that defines a content based search string and another that defines a linked set of documents that each have a hyperlink linkage of a specified range of linkage levels referenced from a target web page, wherein documents in the linked set of documents are characterized in that they can be reached from the target through a sequence of N hyperlink activations, where N is a fixed integer;

coupling a multilevel-search interface signal to a user, the multilevel search interface being configured accept at least two parameters, one that defines a content based search string and another that ~~defines a linked set of documents that each have a hyperlink linkage of a specified range of linkage levels referenced from the target web page;~~ identifies a set of linked set of documents that are linked via one or more levels of hyperlinks to the target web page and fall within the boundaries of a specified domain;

~~generating a set of executable program code and~~ transmitting ~~the~~ a set of executable program code to a remote server for execution on the remote network server, ~~whereby~~ wherein said set of transmitted executable program code orchestrates the following acts:

(i) accessing a first markup language document and scanning said document to determine a hyperlink contained therein;

(ii) determining whether the hyperlink points to a web object whose URL address falls within the boundaries of the specified domain;

~~(ii)~~ (iii) if the hyperlink was found to points to a web object whose URL address falls within the boundaries of the specified domain, activating said hyperlink found in said step of accessing~~, wherein the hyperlink is restricted to be within the specified range N of hyperlinks relative to the target web page~~;

~~(iii)~~ (iv) retrieving at least a portion of a second markup document associated with said hyperlink; and

~~(iv)~~ (v) comparing the contents of said at least a portion of said second markup document to at least a portion of said content based search string.

10. (cancelled)

11. (cancelled)

12. (Previously presented) The method of Claim 9, wherein said content based search string includes a Boolean keyword expression.

13. (Previously presented) The method of Claim 9, wherein said client browser is hosted within a wireless mobile device and said parameter set includes information derived from an electronic positioning system.

14. (Previously presented) The method of Claim 9, whereby said transmitted executable code further orchestrates the following act:

evaluating the results of the comparison and when said step of comparing reveals a match, coupling information related thereto to the user, and when said step of comparing does not yield a match, checking to see if the search is complete, and if it is not, accessing a next hyperlink and repeating the steps of activating, retrieving, and comparing, and evaluating.

15. (Previously presented) The method of Claim 14, wherein said step of evaluating further comprises the steps of:

when said information has been coupled to said user, awaiting a find-next signal, and when said find-next signal is received, checking to see if the search is complete, and if it is not, accessing a next hyperlink and repeating the steps of activating, retrieving, and comparing, and evaluating.

16. (Previously presented) The method of Claim 9, wherein said parameters include a Boolean keyword expression, an indication of the number of levels to search, and an indication to continue the search on a designated-next-linked page.

17. (Previously presented) The method of Claim 9, wherein said hyperlink points to a metadata description of a web resource and said step of accessing involves accessing a file containing metadata relating to said resource.

18. (Previously presented) The method of Claim 9, wherein said second markup document comprises a metadata description, said metadata description being described using a resource description framework (RDF) based language.

19. (Presently Amended) In an intelligent client, a method of seeking information in an information network, the method comprising the steps of:

accessing a web page via said network connection using a client-server transaction;

presenting said web page to a user;

receiving a set of one or more multilevel search parameters to define a multilevel browsing operation over a ~~graph of hyperlinks reachable from said web page in N hops, where N is a positive integer;~~ set of documents whose web addresses fall within a specified domain;

receiving a content based search parameter;

specifying in said intelligent client a set of transportable executable program code, and transmitting the transportable executable program code to a remote server that is adapted to execute the transportable executable program code, said transportable executable program code operative to orchestrate the implementation of said multilevel browsing operation from a remote network node;

wherein the transportable executable program code causes said multilevel browser function to be performed at least partially in said remote server, and wherein the multilevel browser function causes a set of specified links who are within N hyperlink hops of the web page to be searched for the content based search parameter.

20. (Previously Presented) The method of Claim 19, wherein transportable executable program code is represented as Java bytecodes, executes at least partially in an agent sandbox, and uses a remote method invocation based distributed object protocol to communicate with said intelligent client.

21. (New) A smart browser module comprising:

an application layer interface coupled to a protocol stack, said application layer interface operative to receive at least one data packet comprising at least a portion of a target web page;

a user interface for interacting with a user;

a multilevel search control interface configured to accept at least two parameters, one that defines a content based search string and another that defines a linked set of

documents that each have a hyperlink linkage of a specified range of linkage levels referenced from the target web page;

a set of portable executable program code that is responsive to the at least two parameters, and is programmed to perform a multilevel search to search each of the documents specified in the linked set of documents for the content that matches the content based search string, wherein documents in the linked set of documents are characterized in that they can be reached from the target through a sequence of N hyperlink activations, where N is a fixed integer;

a network based program module that causes at least a portion of the portable set of portable executable program code to be transmitted to a server that is adapted to accept executable code and to execute such code on behalf of one or more remote clients;

wherein upon being received at the server, at least some executable code that has been transferred from the browser module is permitted to execute at the server, and upon execution, the portion of the executable code that is executed orchestrates a multilevel search to cause the documents in the linked set of documents to be analyzed to determine whether they match the content based search string.